

The Cluster Finding Scheme for the 2011 FMS Trigger

E.G. Judd
January 7, 2011

Change Log:

Date	Description
January 7, 2011	First version for 2011

A new set of DSM algorithms has been designed and implemented for the FMS detector to use during the 2011 RHIC running period. The previous set of algorithms, implemented for the 2009 run, were optimized to trigger on small, sharply defined clusters of energy. For the 2011 run the aim is to trigger on broader, more diffuse, clusters. This change in emphasis has led to the need for new algorithms. Fortunately the new trigger requirements fit reasonably well into the existing hardware cabling scheme. As a result there will be no changes to any of the hardware, or the QT board algorithms. The only changes are to the DSM algorithms.

FMS Cell to QT Board Assignment Scheme

The FMS cells are assigned to QT boards using a geometrical scheme composed of a large set of simple rectangles. The assignment of FMS cells to the 4 QT8 cards within each QT board has been done using a “striping” scheme, where the stripes run along the length of the rectangles. This scheme is shown in Figure 1.

- All of the FMS cells that connect to the DSM tree are shown in Figure 1.
- The cells contained within a thin solid line are all connected to one QT board.
- Within each QT board the thick solid line marks the locations of clusters that are totally contained by that QT board.
- In 3 of the 4 quadrants the stripes, indicating which QT8 card a cell is assigned to, are shown in light and dark gray.
- In the upper right quadrant the stripes are shown in shades of blue and yellow.
- In this quadrant each QT board has a label from A to J. The 4 stripes in each QT board (representing the 4 QT8 cards) are labeled from 0 to 3. In the rest of this document individual QT8 cards are referred to using these reference numbers, e.g. A(1), E(0), etc....
- Finally, the cells are separated out into 3 groups, indicating which QT board is connected to each of 3 layer-0 DSMS.

The algorithm in the QT boards is split between the QT8 daughter cards and the motherboard. The sum of ADC values from each QT8 card is calculated. In parallel, the cell with the highest ADC value is identified. The user has the option to exclude certain cells from the search for the highest ADC value. This makes it possible to ignore cells that fall on the boundaries as well as dead and noisy cells. A more detailed description will be provided by Chris Perkins.

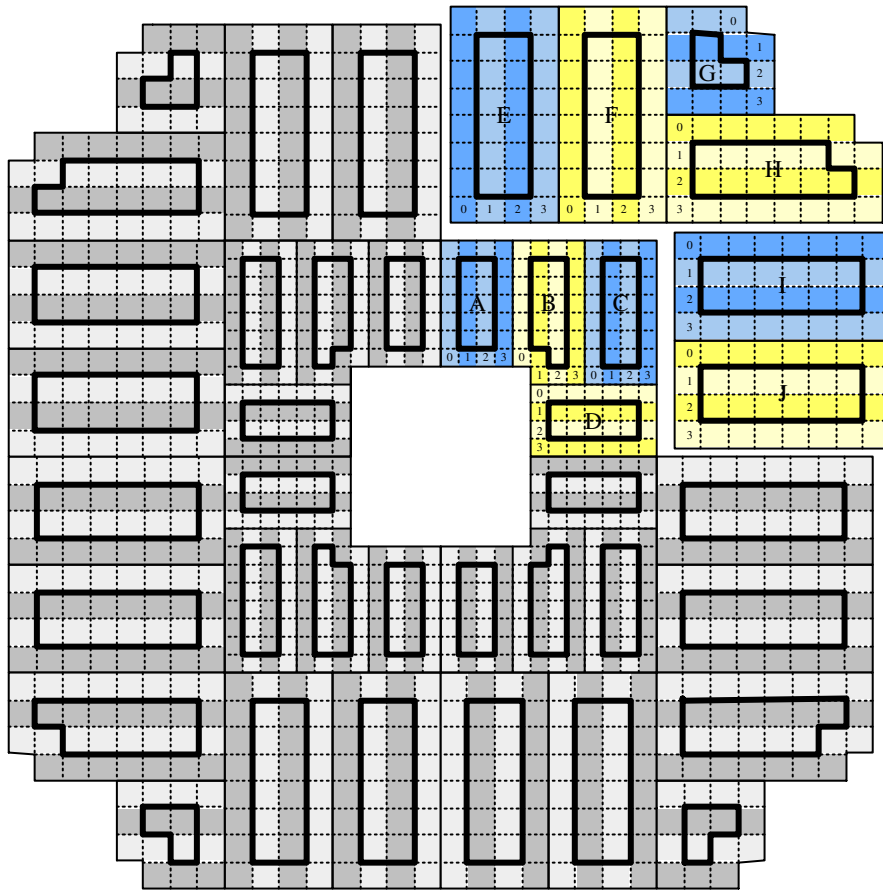


Figure 1: FMS Cell to QT and Layer-0-DSM Assignment Scheme

The output of each QT board, which is sent to the layer 0 DSM boards, contains the following information:

QT8(0:3)	Sum of the ADC values from each of the 4 QT8 cards	4 x 5 bits
HT	ADC value from cell with the highest value	7 bits
HTID	ID of cell with the highest ADC value	5 bits

Within each QT board the numbering scheme used to assign ID values to the 8 channels of each of the four QT8 cards is:

ID	QT8
0:7	0
8:15	1
16:23	2
24:31	3

Layer 0 DSM Tree Algorithms

In the layer 0 DSM boards, clusters are reconstructed by adding together the QT8 sums in various pre-defined groups. A subset of bits from the resulting totals is all that is needed to give the required range and resolution, so the results are truncated, with logic to indicate when an overflow has occurred. Table 1 shows the equations that are used to reconstruct clusters located in any of the cells of the upper-right quadrant of the FMS, i.e. the colored part of Figure 1.

DSM Board	Sum Name	Equations
Small cell quadrant	sumA	$A(0) + A(1) + A(2) + A(3)$
	sumAB	$A(2) + A(3) + B(0) + B(1)$
	sumB	$B(0) + B(1) + B(2) + B(3)$
	sumBC	$B(2) + B(3) + C(0) + C(1)$
	sumC	$C(0) + C(1) + C(2) + C(3)$
	sumD	$D(0) + D(1) + D(2) + D(3)$
Large cell, upper/lower section	sumE	$E(0) + E(1) + E(2) + E(3)$
	sumEF	$E(2) + E(3) + F(0) + F(1)$
	sumF	$F(0) + F(1) + F(2) + F(3)$
	sumG	$G(0) + G(1) + G(2) + G(3)$
	sumGH	$G(2) + G(3) + H(0) + H(1)$
	sumH	$H(0) + H(1) + H(2) + H(3)$
Large cell, side section	sumI	$I(0) + I(1) + I(2) + I(3)$
	sumIJ	$I(2) + I(3) + J(0) + J(1)$
	sumJ	$J(0) + J(1) + J(2) + J(3)$

Table 1: Equations for Cluster Reconstruction in Upper-Right Quadrant of the FMS

In parallel each highest tower (HT) value is compared to two thresholds, and the results for each threshold are OR'ed together.

The DSM hardware imposes a strict limit of 32 bits on the output of each DSM board. The truncated QT8 sums, combined with the HT threshold bits, use all 32 output bits from two of the three DSM boards in each quadrant. This means there are no bits available to pass along individual QT8 sums, or partial sums. As a result there will be no attempt to reconstruct clusters that span the boundaries between layer 0 DSM boards.

There will be 3 slightly different versions of the layer 0 algorithm: one for the DSM board that processes small cell data (QT A-D), a second algorithm for the DSM board that processes the upper/lower section of the large cell array (QT E-H) and a final version for the DSM board that process data from the side of the large cell array (QT I and J). They will operate as follows:

Small-Cell Layer 0 DSM Board: FMS_FM001

- RBT File: fms_fm001_2011_a.rbt
- Users: FM001, FM002, FM003, FM004
- Input: The QT board connections are reversed from what you might expect:
 - Channels 0/1 = QT Board D
 - Channels 2/3 = QT Board C
 - Channels 4/5 = QT Board B
 - Channels 6/7 = QT Board A

From each board the DSM receives:

- Bits 0:4 = QT8(0) sum
- Bits 5:9 = QT8(1) sum
- Bits 10:14 = QT8(2) sum
- Bits 15:19 = QT8(3) sum
- Bits 20:26 = HT
- Bits 27:31 = HTID
- LUT: 1-to-1 mapping
- Registers: Two registers for the HT threshold logic that runs in parallel to the cluster finding logic:
 - R0: FMSsmall-HT-th0 (7 bits)
 - R1: FMSsmall-HT-th1 (7 bits)
- Step 1: Latch the input data.
- Step 2: Compare each HT value to the thresholds specified in R0 and R1. In parallel, add the QT8 sums in pairs, i.e. $A(0) + A(1)$, $A(2) + A(3)$, etc...
- Step 3: Combine (OR) the HT threshold comparison bits for each threshold separately. Use the pair sums to complete the cluster sums listed in Table 1. Extract the 5 least significant bits from each cluster sum. Set the result to 31 (i.e, binary 11111) if either of the two most significant bits is set.
- Step 4: Latch the output data:
 - Bits 0:4 = sumD
 - Bits 5:9 = sumC
 - Bits 10:14 = sumBC
 - Bits 15:19 = sumB
 - Bits 20:24 = sumAB
 - Bits 25:29 = sumA
 - Bit 30 = HT threshold 0
 - Bit 31 = HT threshold 1

Large-Cell Layer 0 DSM Board: FMS_FM005 (Upper/Lower section of the array)

- RBT File: fms_fm005_2011_a.rbt
- Users: FM005, FM007, FM009, FM011
- Input: The QT board connections are reversed from what you might expect:
 - Channels 0/1 = QT Board H
 - Channels 2/3 = QT Board G
 - Channels 4/5 = QT Board F
 - Channels 6/7 = QT Board E

From each board the DSM receives:

- Bits 0:4 = QT8(0) sum
- Bits 5:9 = QT8(1) sum
- Bits 10:14 = QT8(2) sum
- Bits 15:19 = QT8(3) sum
- Bits 20:26 = HT
- Bits 27:31 = HTID
- LUT: 1-to-1 mapping
- Registers: Two registers for the HT threshold logic that runs in parallel to the cluster finding logic:
 - R0: FMSlarge-HT-th0 (7 bits)
 - R1: FMSlarge-HT-th1 (7 bits)
- Step 1: Latch the input data.
- Step 2: Compare each HT value to the thresholds specified in R0 and R1. In parallel, add the QT8 sums in pairs, i.e. $E(0) + E(1)$, $E(2) + E(3)$, etc...
- Step 3: Combine (OR) the HT threshold comparison bits for each threshold separately. Use the pair sums to complete the cluster sums listed in Table 1. Extract the 5 least significant bits from each cluster sum. Set the result to 31 (i.e, binary 11111) if either of the two most significant bits is set.
- Step 4: Latch the output data:
 - Bits 0:4 = sumH
 - Bits 5:9 = sumGH
 - Bits 10:14 = sumG
 - Bits 15:19 = sumF
 - Bits 20:24 = sumEF
 - Bits 25:29 = sumE
 - Bit 30 = HT threshold 0
 - Bit 31 = HT threshold 1

Large-Cell Layer 0 DSM Board: FMS_FM006 (Side section of the array)

- RBT File: fms_fm006_2011_a.rbt
- Users: FM006, FM008, FM010, FM012
- Input: The QT board connections are reversed from what you might expect:
 - Channels 0/1 = QT Board J
 - Channels 2/3 = QT Board I
 - Channels 4/5 = Unused
 - Channels 6/7 = Unused

From each board the DSM receives:

- Bits 0:4 = QT8(0) sum
- Bits 5:9 = QT8(1) sum
- Bits 10:14 = QT8(2) sum
- Bits 15:19 = QT8(3) sum
- Bits 20:26 = HT
- Bits 27:31 = HTID
- LUT: 1-to-1 mapping

- Registers: Two registers for the HT threshold logic that runs in parallel to the cluster finding logic:
 - R0: FMSlarge-HT-th0 (7 bits)
 - R1: FMSlarge-HT-th1 (7 bits)
- Step 1: Latch the input data.
- Step 2: Compare each HT value to the thresholds specified in R0 and R1. In parallel, add the QT8 sums in pairs, i.e. $I(0) + I(1)$, $I(2) + I(3)$, etc...
- Step 3: Combine (OR) the HT threshold comparison bits for each threshold separately. Use the pair sums to complete the cluster sums listed in Table 1. Extract the 5 least significant bits from each cluster sum. Set the result to 31 (i.e, binary 11111) if either of the two most significant bits is set.
- Step 4: Latch the output data:
 - Bits 0:14 = Unused
 - Bits 15:19 = sumJ
 - Bits 20:24 = sumIJ
 - Bits 25:29 = sumI
 - Bit 30 = HT threshold 0
 - Bit 31 = HT threshold 1

Layer 1 DSM Tree Configuration and Algorithm

From Figure 1 it can be seen that there are 3 types of layer 0 DSM board: small cell quadrant, large cell upper/lower section and large cell side section. There are 4 layer 0 DSM boards of each type, one in each of four quadrants. This makes 12 layer 0 DSM boards in total. Since there are no plans to trigger on clusters that span the boundary between large and small cells there is no immediate need to bring the large and small cell data together in the DSM tree. By the same logic, there is no need to bring the data from the North and South sides of the large-cell array together. The connections from the layer 0 DSM boards to the layer 1 DSM boards have therefore been made as simple and as tightly packed as possible. The 4 boards covering the small cells will all connect to one layer 1 DSM board: FM101. The 4 boards covering the South side of the large cell array will connect to a second layer 1 DSM board (FM102), and the 4 boards covering the North side of the array will connect to the third and final layer 1 DSM board (FM103). This is shown in Figure 2.

The layer 1 algorithms will apply three thresholds to each cluster sum and OR the results for each of the three thresholds separately. In addition, every layer 1 DSM will calculate a total energy sum for each of the quadrants that it covers. The results will be truncated to 6 bits, with logic to indicate when an overflow has occurred. In parallel the HT threshold bits will be OR'ed together.

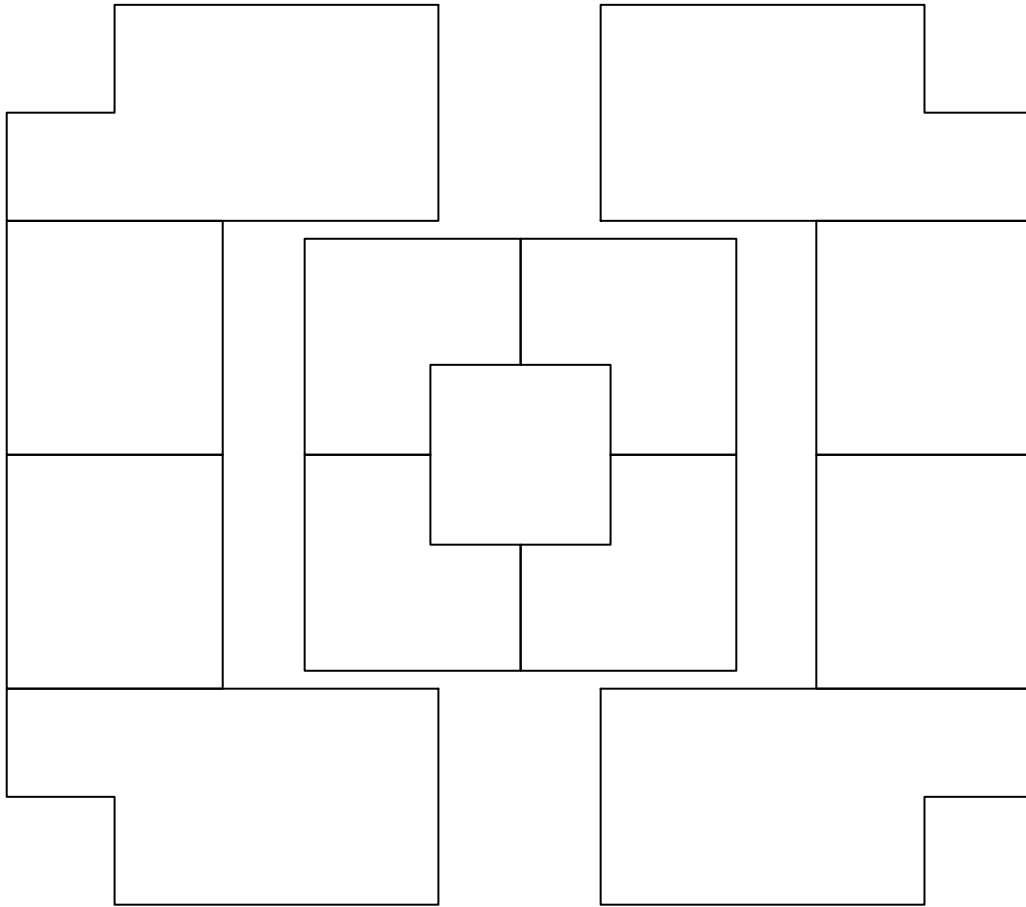


Figure 2: DSM Layer 0 to Layer 1 Assignment Scheme

The layer 1 DSM algorithms will therefore perform the following steps:

Small-Cell Layer 1 DSM Board: FMS_FM101

- RBT File: fms_fm101_2011_a.rbt
- Users: FM101
- Input:
 - Channels 0/1 = FM001, South-Top
 - Channels 2/3 = FM002, South-Bottom
 - Channels 4/5 = FM003, North-Top
 - Channels 6/7 = FM004, North-Bottom

From each board the DSM receives:

- Bits 0:4 = sumD
- Bits 5:9 = sumC
- Bits 10:14 = sumBC
- Bits 15:19 = sumB
- Bits 20:24 = sumAB
- Bits 25:29 = sumA

- Bit 30 = HT threshold 0
 - Bit 31 = HT threshold 1
- LUT: 1-to-1 mapping
- Registers: Three 5-bit registers for applying thresholds to the cluster sums:
 - R0: FMSsmall-cluster-th0
 - R1: FMSsmall-cluster-th1
 - R2: FMSsmall-cluster-th2
- Step 1: Latch the input data.
- Step 2: Combine (OR) the HT threshold bits for the South side, and separately for the North side. Compare each of the 24 input 5-bit sums to each of the three cluster thresholds. For each quadrant, make the following two sums: sumA+sumB and sumC+sumD
- Step 3: Combine (OR) the HT threshold bits from the South and North sides. Combine (OR) the cluster threshold bits. Add together the pair sums to complete the quadrant sums for each quadrant: (sumA+sumB) + (sumC+sumD). Extract the 6 least significant bits from each quadrant sum. Set the result to 63 (i.e, binary 111111) if the most significant bit is set.
- Step 4: Latch the output data:
 - Bits 0:5 = South-Top (ST) quadrant sum
 - Bits 6:11 = South-Bottom (SB) quadrant sum
 - Bits 12:17 = North-Top (NT) quadrant sum
 - Bits 18:23 = North-Bottom (NB) quadrant sum
 - Bits 24:26 = Cluster threshold bits from full small cell array
 - Bits 27:28 = HT threshold bits from full small cell array

Large-Cell Layer 1 DSM Board: FMS_FM102

- RBT File: fms_fm102_2011_a.rbt
- Users: FM102 (South), FM103 (North)
- Input:
 - Channels 0/1 = Top (FM005 and FM009)
 - Channels 2/3 = Upper-Side (FM006 and FM010)
 - Channels 4/5 = Bottom (FM007 and FM011)
 - Channels 6/7 = Lower-Side (FM008 and FM012)

From the Top and Bottom boards the DSM receives:

- Bits 0:4 = sumH
- Bits 5:9 = sumGH
- Bits 10:14 = sumG
- Bits 15:19 = sumF
- Bits 20:24 = sumEF
- Bits 25:29 = sumE
- Bit 30 = HT threshold 0
- Bit 31 = HT threshold 1

From the Side boards the DSM receives:

- Bits 0:14 = Unused
- Bits 15:19 = sumJ
- Bits 20:24 = sumIJ

- Bits 25:29 = sumI
 - Bit 30 = HT threshold 0
 - Bit 31 = HT threshold 1
- LUT: 1-to-1 mapping
- Registers: Three 5-bit registers for applying thresholds to the cluster sums:
 - R0: FMSlarge-cluster-th0
 - R1: FMSlarge-cluster-th1
 - R2: FMSlarge-cluster-th2
- Step 1: Latch the input data.
- Step 2: Combine (OR) the HT threshold bits for the upper quadrant, and separately for the lower quadrant. Compare each of the 18 input 5-bit sums to each of the three cluster thresholds. For each quadrant, make the following two sums: sumE+sumF+sumG and sumH+sumI+sumJ
- Step 3: Combine (OR) the HT threshold bits from the upper and lower quadrants. Combine (OR) the cluster threshold bits. Add together the tripple sums to complete the quadrant sums for each quadrant: (sumE+sumF+sumG) + (sumH+sumI+sumJ). Extract the 6 least significant bits from each quadrant sum. Set the result to 63 (i.e, binary 111111) if the most significant bit is set.
- Step 4: Latch the output data:
 - Bits 0:5 = Top quadrant sum
 - Bits 6:11 = Bottom quadrant sum
 - Bits 12:23 = Unused
 - Bits 24:26 = Cluster threshold bits
 - Bits 28:27 = HT threshold bits

FPD-East QT Algorithm

In parallel with the FMS, the FPD-East detector will also be taking data. The electronics chain involves 4 QT boards passing data to a layer 1 DSM board, which then sends its output to a common layer 2 DSM board, shared with the FMS. The QT board algorithm will add all 32 12-bit ADC values to produce a 17-bit sum. The user has the option, through the use of a register-settable mask, to exclude certain channels from the sum. This makes it possible to ignore dead or noisy cells. A more detailed description will be provided by Chris Perkins.

FPD-East Layer 1 DSM Algorithm

The layer 1 algorithm for FPD-East will combine the 4 input sums from the 4 QT boards to make two 18-bit module sums. Each of those sums will be compared to a threshold, and the threshold bits will be passed on to layer 2. The details of the FPD-East layer 1 DSM algorithm are as follows:

FPD-East Layer 1 DSM Board: MIX_FE101

- RBT File: mix_fe101_2009_a.rbt
- Users: FE101
- Input:

- Channels 0/1 = Section 1 (FEQ-QT1)
- Channels 2/3 = Section 2 (FEQ-QT2)
- Channels 4/5 = Section 3 (FEQ-QT3)
- Channels 6/7 = Section 4 (FEQ-QT4)

From each board the DSM receives:

- Bits 0:16 = Sum of 32 12-bit ADC values
- LUT: 1-to-1 mapping
- Registers: Two registers that are combined to make one 18-bit threshold value.
NOTE: the maximum size of any register is 16 bits, which is too small for an 18-bit value.
 - R0: FPE-threshold-12LSB
 - R1: FPE-threshold-6MSB
- Step 1: Latch the input data.
- Step 2: Add the 17-bit input sums from QT1 and QT2 to produce the 18-bit sum for module 1. Add the 17-bit input sums from QT3 and QT4 to produce the 18-bit sum for module 2.
- Step 3: Compare each of the two module sums to the 18-bit threshold value formed by concatenating the values specified in R0 and R1.
- Step 4: Latch the output data:
 - Bit 0 = Threshold bit from module 1
 - Bit 1 = Threshold bit from module 2

Layer 2 DSM Algorithm

The layer 2 DSM board will receive data from the FMS small and large cell layer 1 DSMS and the FPE layer 1 DSM. The algorithm will combine the FMS threshold data from the quadrants to give one set of HT threshold data covering the entire array, along with cluster threshold data covering the small and large cell parts of the array separately. In parallel, the sums from the small and large cells will be combined for each quadrant to get a full quadrant sum, known as a jet patch. Those jet patch sums will be compared to three thresholds, and the threshold bits will be combined (OR). In addition, the number of quadrants with the first cluster threshold bit set will be counted. This will provide a way to trigger on multi-cluster events. The two FPE threshold bits will just be combined (OR) together. Therefore, the layer 2 algorithm will operate as follows:

- RBT File: 11_fp201_2011_a.rbt
- Users: FP201
- Input:
 - Channels 0/1 = FM101, FMS Small cells
 - Channels 2/3 = FM102, FMS Large cells, South side
 - Channels 4/5 = FM103, FMS Large cells, North side
 - Channel 6 = Unused
 - Channel 7 = FE101, FPE

From the FMS small cell DSM the input is:

- Bits 0:5 = South-Top (ST) quadrant sum

- Bits 6:11 = South-Bottom (SB) quadrant sum
- Bits 12:17 = North-Top (NT) quadrant sum
- Bits 18:23 = North-Bottom (NB) quadrant sum
- Bits 24:26 = Cluster threshold bits from full small cell array
- Bits 27:28 = HT threshold bits from full small cell array

From the FMS large cell DSMs the input is:

- Bits 0:5 = Top quadrant sum
- Bits 6:11 = Bottom quadrant sum
- Bits 12:23 = Unused
- Bits 24:26 = Cluster threshold bits
- Bits 28:27 = HT threshold bits

From the FPE DSM the input is:

- Bit 0 = FPE module 1 threshold bit
- Bit 1 = FPE module 2 threshold bit

- LUT: 1-to-1 mapping
- Registers: Three 7-bit registers for applying thresholds to the jet patches
 - R0: FMS-JP0
 - R1: FMS-JP1
 - R2: FMS-JP2
- Step 1: Latch the input data.
- Step 2: Combine (OR) the two FPE bits. Combine (OR) the HT bits from the North and South sides of the FMS large cell array. Do the same for their cluster bits. Delay the HT bits from the FMS small cell array to the 3rd step, and delay the associated Cluster bits to the 4th step. Make the jet patches by adding together the small and large cell sums for each quadrant, e.g.
 - $JP-ST = ST-small + ST-large$
- Step 3: Delay the FPE bit to Step 4. Combine (OR) the HT bits from the FMS small and large cell arrays. Delay the FMS large cell cluster bits to the 4th step. Compare each of the four jet patches to the three thresholds and combine (OR) the results for each threshold. In parallel count how many patches exceed the JP0 threshold, and set the dijet but if that number is greater than 1.
- Step 4: Latch the output data. The output data format is shown in Table 4 (next page). In parallel, send a copy of this set of bits to the scaler system.

Table 4: Output of Layer 2 FPD/FMS DSM Board

Bit	Name	Description
Bit 0	Fms-HT-th0	FMS small-cell HT threshold-0 bit
Bit 1	Fms-HT-th1	FMS large-cell HT threshold-1 bit
Bit 2	FmsSml-Cluster-th0	FMS small-cell cluster threshold-0 bit
Bit 3	FmsSml-Cluster-th1	FMS small-cell cluster threshold-1 bit
Bit 4	FmsSml-Cluster-th2	FMS small-cell cluster threshold-2 bit
Bit 5	FmsLrg-Cluster-th0	FMS large-cell cluster threshold-0 bit
Bit 6	FmsLrg-Cluster-th1	FMS large-cell cluster threshold-1 bit
Bit 7	FmsLrg-Cluster-th2	FMS large-cell cluster threshold-2 bit
Bit 8	Fms-JP-th0	FMS Jet Patch threshold-0 bit
Bit 9	Fms-JP-th1	FMS Jet Patch threshold-1 bit
Bit 10	Fms-JP-th2	FMS Jet Patch threshold-2 bit
Bit 11	Fms-dijet	FMS dijet bit
Bit 12	Unused	Unused
Bit 13	Unused	Unused
Bit 14	FPE	FPE trigger bit
Bit 15	Unused	Unused